

# OFFICE COPY

## Apple — Aids

DISK - BASED PROGRAMMING UTILITY SYSTEM

32K  
or  
48K



DOS  
3.3

FORMAT BLANK DISK ★ KILL DOS ★ DISK MAP  
IMPROVED DIRECTORY ★ SECTOR LISTINGS  
UNDELETE FILES ★ DISK EDITOR ★ DISK COPY  
CREATE EXEC FILES ★ EDIT EXEC FILES  
BASE CONVERT

\$34.95  
SUGGESTED  
RETAIL



**Discovery  
Software**

**P.O. Box 1748  
Highland, IN 46322**



## APPLE-AIDS

Apple-Aids is Copyright 1981 by  
Discovery Software - All commercial  
rights reserved.

Apple is a registered trademark of  
Apple Computer, Inc.



APPLE AIDS  
Table of Contents

Subject/Program	Page
0) Introduction	2
1) Format Blank Disk	4
2) Kill DOS	6
3) Disk Map	7
4) Improved Directory	9
5) Sector Listings	10
6) Undelete Files	11
7) Disk Editor	13
8) Disk Copy	17
9) Create Exec Files	18
10) Edit Exec Files	19
11) Base Convert	20
12) Exit System	21
13) Getting started	22
14) User Notes	24



Chapter 0

INTRODUCTION

THANK YOU FOR CHOOSING THIS PRODUCT !

The software you have chosen is the product of many hours of strenuous work on the part of many people. If you enjoy the use of this product, please recommend it to your friends. We would be honored by such a recommendation, but we ask that you do not copy this diskette for any purpose other than your own back-up usage. If you do copy and distribute this diskette without the express written permission of Discovery Software, you are not only breaking the law, but you are depriving others of their just income. If this product is good enough for you to purchase, it is surely good enough for your friends to purchase.

Copyright Discovery Software, 1981  
all rights reserved

This product is sold on an "as is" basis, and although we believe this software package to be accurate, no representation is made or is to be implied as to the suitability, accuracy or fitness of any of the programs for any particular purpose. Discovery Software shall not be held liable for any damages, incidental or consequential, resulting from the use, misuse, performance, or supply of these programs.

All of our programs have been

thoroughly tested and de-bugged. However, occasionally we do miss something. If you feel that you have discovered something that we have missed, please take a minute to let us know. Our address is:

DISCOVERY SOFTWARE  
P.O. Box 1748  
Highland, IN 46322  
(219) 924-5549

Also, be sure to return your user registration card. You will then be notified of new software additions to our quality line of products, and we, in turn, will better be able to help you.

AS A FINAL NOTE, PLEASE READ CHAPTER 13 "GETTING STARTED" BEFORE ATTEMPTING TO USE APPLE-AIDS. IT MAY SAVE YOU SOME TIME.



## Chapter 1

## FORMAT BLANK DISK

When you format a disk using the "INIT" command of DOS, your Apple writes a series of identifying marks on the diskette so that it can find it's way around when it needs to. After doing this, the disk is left completely blank (for a short time). At this point there are 35 tracks that have been written, each consisting of 16 sectors. This totals (by simple math), to 560 blank sectors. Then "INIT" continues and marks off 64 sectors for internal usage. Tracks 0,1, and 2 are used for an image of the Disk Operating System, and track 17 (\$11) is used for the VTOC and the CATALOG. Then the "HELLO" program you have specified is linked and saved so that it runs when you boot DOS. All this leaves you 64 sectors short of full disk usage, however.

This program will format your diskette in a slightly different manner. What happens is that the disk is marked into tracks and sectors, and no image of DOS is saved on the disk. However, track 0 and 17 (\$11) are still "off limits" to a user. Track 0 is not accessible (except when booting or seeking directly, as with the Disk Editor), and track 17 (\$11) is still required so that DOS can keep track of what you have saved on the disk.

To run the program, choose option 1 from the Main Program Menu. Then decide which slot and drive you wish

to format from, and follow the instructions.

An important thing to remember is that you will not be able to boot your system using this disk, since there is no DOS image on the disk. Other than that, the disk will behave and appear as normal, except that the storage capacity has been increased by 32 sectors.

## Chapter 2

## KILL DOS

"Kill DOS" will allow you to accomplish the same objective as "Format Blank Disk", but in a slightly different manner. This program assumes that you already have a normally formatted diskette and that you have saved things to the disk already. In fact, you may have saved so much to the disk that you have found that you have run out of room to store some data you need.

Never fear! "Kill DOS" will allow you to take DOS off of your diskette without harming so much as a single byte of your valuable information. Again, this will "free-up" 32 sectors for your use, but you will not be able to boot your system using your disk (on the Apple, it is a pretty good trick to be able to boot without something to boot with).

Once you have removed DOS, you cannot restore it without re-initializing the diskette, and hence loosing what was on the disk. So make sure you want to "Kill DOS" before you actually do it.

To run this function, simply choose option 1 from the Main Program Menu, and determine which slot and drive you wish to perform the operation on. If you have more than one drive on your system, take care that you are "killing" the DOS on the correct drive.

## Chapter 3

## DISK MAP

Many times, especially when doing highly specialized application software, you need to be able to see exactly which sectors are in use or free on a given diskette. "Disk Map" allows you to determine this information from any normally formatted diskette.

To use this utility, choose option 3 from the Main Program Menu. Then simply determine the slot and drive you wish to map, and press RETURN when so prompted. Be patient, it takes a few short seconds to completely map the disk.

Now you will see a graphic representation of the space allocation on the disk. In use sectors will be denoted with an asterisk (\*). On this graph, tracks are represented on the horizontal axis, and sectors on the vertical. Hence, if you wished to determine if track 13 (\$D), sector 15 (\$F) was in use, you would locate the track across the top of the chart, and the sector along the side. If there is an asterisk (\*) where the row and column meet, then the sector is in use.

"Disk Map" also keeps count for you, and will let you know total sectors free as well as total sectors in use. This is handy to determine if you have enough room on a disk to transfer a file completely to the disk. These

totals can be found at the bottom of the screen.

To end the mapping session, simply press RETURN, and you will be returned to the Main Program Menu.

## Chapter 4

## IMPROVED DIRECTORY

How many times have you wished you knew exactly what was on a disk's catalog. Improved Directory will let you see this information. When you have chosen the slot and drive you wish, press RETURN and you will be shown the file name (truncated at 10 characters), the file type (A=Applesoft, I=Integer, B=Binary, T=Text), and the file's code.

The code of a file may be any of three states: unlocked, locked, or deleted. a blank, or no mark, denotes that the file is unlocked. an asterisk (\*) denotes the file is locked, and a dollar sign (\$) denotes that the file is deleted.

This may appear a bit baffling at first, but DOS does not erase a file name from the directory simply because it has been deleted. All it does is to mark it as deleted so that it will not appear when you use the command "CATALOG". It also assures that next time you save something to the disk, it will "replace" the catalog entry you previously deleted.

Besides the code, if the file type is 'B' (Binary), you will be shown the beginning address and file length of the program. Both of these numbers are in decimal notation. This can be very helpful, as this information is not shown with the normal "CATALOG" function of DOS.

## Chapter 5

### SECTOR LISTINGS

Sector Listings come in very handy if you wish to pin-point where (exactly) a file resides on disk.

When you answer the usual questions as to slot and drive, and then press RETURN, you will be shown a series of numbers in the format (TT-SS). In other words, the track is shown first, and then the sector, so that if you see the numbers (17-15), part of the file you are examining is contained on track 17, sector 15.

The numbers shown in inverse (black on white) denote the sector used for the track-sector list of the file being examined. All other sectors are shown in normal mode (white on black).

Also, it is handy to remember that all information shown is represented in decimal notation.

## Chapter 6

## UNDELETE FILES

I don't know about you, but there are many times when I have deleted files that I didn't want to, but once I hit return, it was too late. This has usually happened late at night when I have been up for 43 hours straight and am having trouble seeing the monitor, let alone hitting the right keys.

Be that as it may, I am left with no way to recover a file I just deleted, right?

Wrong! All you need to do is to use "Undelete Files" to salvage that lost program, and it will magically reappear in complete form. Perhaps a few words about how DOS deletes files is in order here.

First of all, DOS, upon receipt of the "DELETE" command, will mark the directory entry for the file by moving relative byte 0 to relative byte 32 (\$20) and setting relative byte 0 to the value 255 (\$FF). Now the file will not show when you ask for a "CATALOG" of the disk. But this is not all, for now DOS updates the track bit map for any tracks that the file may have been saved on. Now the file is gone, because next time you save something, it will be written into the space previously occupied by the file you just deleted.

This program sets relative byte 0 of the Directory back to it's original



value, and puts the track bit map back to it's "pre-deleted" values. Now the file is ressurected and all is as before.

There are a few pointers to remember, however. If you have deleted a program and later decide to undelete it, this will only be meaningful if you have not saved anything else out to the disk since you last deleted the file. If you did, it is very likely that DOS, thinking you meant it when you deleted the file, has written your new file into part, if not all, of the space previously occupied by the deleted file. If you try to undelete under these circumstances, chances are you will mess up the directory, and that will mean extensive patch work later on, a blown directory, or garbage files. In any case, it is a good practice to use "Undelete Files" only immediately after having deleted the file.

## Chapter 7

### DISK EDITOR

This is the single most used program in our library. Disk Editor is a powerful utility that will allow you to load, inspect, alter, and write any sector of any normally formatted disk.

Choose option 7 from the Main Program Menu and answer the questions about which slot and drive you will be using. Now you will see a prompt asking you for a command. The commands and their actions are:

L) This command will LOAD any sector you designate into the editor buffer and then display it on the screen in the current display mode (for display modes, see commands A, D, and H).

W) This is the opposite of load, i.e. this command will WRITE the sector currently in the editor buffer to any place you designate on the disk.

E) EDIT. This will allow you to change any or all of the information displayed on the screen. These changes are then made in the editor buffer and can then be written back out to the disk. The editing begins at whatever byte you specify, and you have the choice of several edit types. the edit types are:

D) Edit will be a one byte DECIMAL change.

H) Edit will be a one byte HEXADECIMAL change.

A) Edit will be an ASCII string change starting with the specified byte. For instance, with this type of edit you can enter word as the change, and it will be inserted beginning with the starting byte you specified.

E) Edit will be an ENHANCED ASCII string change. This is the same as an Ascii change, except that the character values have the high byte on (Ascii value + 128).

F) Edit mode will be "FILL". You will be asked for the ending byte and the type of fill (decimal or hex), and then the value to fill with. All bytes between the limits you have set will then be changed to the desired value.

All edits are done immediately and then displayed in the current display mode.

+ ) This command will allow you to INCREMENT the sector counter by one, LOAD and DISPLAY in the current mode. Thus if you are currently viewing track 3 sector 4 on the display, upon pressing '+', track 3 sector 5 will be loaded and displayed. Also, track borders are automatically bridged, so that if you are on track 34 (\$22) sector 15 (\$F) and you press "+", track 0 sector 0 will be loaded and displayed.

-) The '-' command works the same

as the '+' command, but in reverse. Pressing '-' will DECREMENT the sector counter by one, LOAD and DISPLAY in the current mode. Track borders are automatically bridged here, also.

S) This command SWITCHES the part of the sector being viewed on the display without changing the display mode. Each page of display can accommodate 128 bytes of data. This is half of the actual sector. To see the other half, press 'S'.

A) This displays the current page of the editor buffer in ASCII representation. All control range characters that would normally not show are denoted by inverse (black on white) print. This display mode is maintained until another display mode is chosen.

D) This displays the current page of the editor buffer in DECIMAL notation. This display mode is maintained until another display mode is chosen.

H) This displays the current page of the editor buffer in HEXADECIMAL notation. This display mode is maintained until another display mode is chosen.

?) This will show you a LIST OF THE COMMANDS AND EDIT TYPES available. To get back to normal display, simply press any of the display mode commands (A, D, or H).

X) Use this to EXIT the program back to the Main Program Menu.

A word on general system format. All byte markers are shown to the left side of the screen in decimal notation. These refer to the first byte of the row. There are eight bytes per row, sixteen rows per page, two pages per sector. When asked for track or sector desired, only decimal input is allowed. On editing, Ascii, decimal, or hex are allowed, depending upon the edit type chosen.

You will notice status markers at the top of the editing screen. These show you which slot and drive you are functioning on, and which track and sector is currently in the buffer. If you wish to write a sector back out to the same place that you loaded it from, just enter the same track and sector as you see noted by the indicators.

If you wish to load a sector from one disk and write it to another disk, simply load it in using the "L" command, and then put the disk to be written to in the drive you are using. Then use the "W" command to put the sector in the buffer wherever you want it on the new disk.

## Chapter 8

## DISK COPY

This program will copy a disk to another disk. Either one or two disk drives can be used, and there is no need to format the disk being copied onto. This program does that automatically.

All you need do is to choose option 8 from the Main Program Menu. Then determine which drive the original (master) disk will be in, and which drive the copy (blank) disk will be in. When you press RETURN the disk denoted as the copy (blank) will be formatted and copied to track by track.

You will notice that you are kept constantly informed as to the status of the copy procedure. At the successful conclusion of the copy, you will be given the chance to make another copy (you can reset parameters at this time), or to end. If you end, you will be returned to the Main Program Menu.

## Chapter 9

### CREATE EXEC FILES

Creating exec files can be a headache sometimes. This utility will allow you to put together a series of commands into a file to be later executed.

First you will be asked for the name of the file. Enter any valid file name. It can contain any characters except a comma (,) or colon (:), and it cannot begin with a number. It also cannot be over 29 characters in length.

Now all you need to do is to enter the commands just as you would at command level, and they will be saved out to the disk. Commas, colons, and quote marks are allowed here, so you can type just about anything you want.

In order to end a creating session, type the word 'QUIT' as your command. It must be the only word on the line. You will then be returned to the Main Program Menu.

## Chapter 10

### EDIT EXEC FILES

Most of the time, it is impossible to get an exec file correct the first time through. It can be very frustrating and tedious to retype the entire file just to correct one mistake.

"Edit Exec Files" will allow you to load, inspect, change, and re-write any exec file on a disk. All you need do is provide the name of the file and any changes you want.

This is a line oriented editor, so if you are changing a line, you must retype the entire line. You can add lines to the end of the file, or you can delete lines from anyplace within the file. All you need to do is to choose the line number you wish to edit.

In order to end editing, enter 'QUIT' as your command, just as in "Creating Exec Files".



## Chapter 11

### BASE CONVERT

Base Convert works on the premise that it is easier to let the computer do what it was intended to do than to do it yourself. This program works with binary, octal, decimal, and hexadecimal numbers. It will convert a number from any of the above bases to all of the bases.

All you need to do is to enter the base code and then the number in the format "CN". The 'C' is the base code, and the 'N' is the number to be converted.

For example, if you wish to convert the decimal number 3456, enter 'D3456' and the computer will respond with:

```
BINARY: 110110000000
OCTAL:  6600
DECIMAL: 3456
HEX:    D80
```

Numbers can be converted from any of the other bases by simply using a different base code: "B" for Binary, "O" for Octal, "D" for Decimal, or "H" for Hexadecimal.

In order to exit the program, simply enter the word "END" when prompted for input, and you will be returned to the Main Program Menu.

## Chapter 12

### EXIT SYSTEM

This is the proper way to exit Apple-Aids. It should be noted here that Apple-Aids uses a modified DOS, and pressing "ctrl-c" will only result in the system hanging indefinitely. Hitting "reset" will result in the system rebooting, and the only way to get back to normal DOS when done with the system is to reboot to another standard system disk. This can be done using "PR#6" or by hitting reset with another disk in drive 1.

## Chapter 13

### GETTING STARTED

In order to use Apple-Aids, you must have either a 32K or a 48K Apple II computer. You must also have at least one disk drive.

Apple-Aids is written in Applesoft and Machine Language, and operates under a modified DOS. In order to use Apple-Aids properly, you must boot your system up using your Apple-Aids disk. You should execute the following steps:

- 1) Insert your Apple-Aids disk into drive 1.

- 2) Turn your computer power on, or type "PR#s" ('s' is the slot containing the controller card of the drive you wish to boot from) and press RETURN.

- 3) You will see the Apple-Aids Main Program Menu appear on your monitor.

If the above three steps fail, check all of your electrical connections (including P.C. board contacts) and repeat the steps again.

After you see the Main Program Menu, it would be a good idea to make a back-up copy of your Apple-Aids program disk. You can do this by choosing option 8 from the menu. Chapter 8 in this manual explains making back-up copies in some more

detail.

Apple-Aids will work with any Apple II computer that has Applesoft in ROM or a Language Card. APPLE-AIDS WILL NOT WORK IF YOU ONLY HAVE APPLESOFT IN RAM!

If all else fails, contact your dealer or give us a call. You will find our number at the front of this manual.

## Chapter 14

### USER NOTES

1) All of Discovery Software's programs are extensively tested and de- bugged. However, you are the final judge. If you feel that you have discovered something that should not be happening (i.e. a "bug"), then WRITE DOWN the conditions and steps that led you to the discovery and apparent problem. Then send us a note or give us a call, and we will try to figure out what happened.

2) All of Discovery Software's programs are covered by our limited warranty. We guarantee them to be free from operational defects. Further information on our guarantee is contained in the introduction to this manual.

3) The only way that our guarantee will apply is if you fill out and return the user registration card contained in this package. It lets us know that you are indeed the owner of this software package.

4) So that we can continue with our fine line of software products, we ask that you fill out and return the user feedback card contained in this package.





